QGIS Application - Bug report #9148
Data Defined Label Locations NULL value doesn't work as expected with SQL Server Layer/PostGIS

2013-12-01 03:49 PM - Alex Leith

Status:	Closed		
Priority:	Severe/Regression		
Assignee:	Martin Dobias		
Category:	Labelling		
Affected QGIS version:	master	**Regression?:**	No
Operating System:		**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	17781

Description

Steps to reproduce issue:

1. Set up a layer in SQL Server with columns for LabelX and LabelY as float with NULL as default value.
2. Note that when labels are turned on with data defined coordinates using those columns, labels are not shown for those features.

If you set up an initial value for the LabelX and LabelY then they work fine and you can move them and save.

I tested the same layer I found the issues with after exporting to SHP and it worked fine, NULL values result in labelling engine label locations.

Associated revisions

Revision 863a83b3 - 2014-06-11 08:08 AM - Martin Dobias

Fix #9148 (data-defined label positions with postgis)

A bit strangely, conversion of null variant with double type
to a double value is considered ok by QVariant

History

#1 - 2013-12-01 04:34 PM - Alex Leith

Found a work-around: if you use a varchar to store the LabelX and LabelY then QGIS behaves as expected.

#2 - 2014-01-28 08:16 AM - Alexandre Neto

- Status changed from Open to Closed

- Target version changed from Version 2.0.0 to Future Release - High Priority

Can can confirm this also with postgis. If the X and Y values are NULL (Label not fixed), the labels are draw near 0,0.

I'm using X and Y as numerical fields with size 20 precision 5.

#3 - 2014-01-28 08:19 AM - Alexandre Neto

- Status changed from Closed to Reopened

#4 - 2014-01-29 11:37 AM - Giovanni Manghi

- Affected QGIS version changed from 2.0.1 to master

- Status changed from Reopened to Feedback

Hi Alexandre, I can confirm the issue, but I'm not sure is a regression. Did it worked ok on previous QGIS releases?

#5 - 2014-01-30 07:03 AM - Alexandre Neto

Hi Giovanni, yes it works well in the stable version (2.0.1). Labels of features with NULL values in X and Y fields are considered not fixed and are displayed in the default place.

#6 - 2014-01-31 06:54 AM - Regis Haubourg

Same behaviour when using memory data provider. Strangely, when creating memory layer, all goes well, but when loading it with memory layer saver when opening projects, Labels without XY disappear.
Confirmed on 2.0.1 and master.

#7 - 2014-01-31 07:01 AM - Regis Haubourg

With memory provider, there is no workaround with varchar.
Pretty serious issue since it ruins the intersest of plugin EasyCustomLabeling. it works once, but when reloading the project, all unpinned labels are masked

#8 - 2014-01-31 07:10 AM - Regis Haubourg

I guess everything is around NULL handling. On my memory layer working well, NULL are in fact objects <class 'PyQt4.QtCore.QPyNullVariant'>. When restored with memLayerSaver, I get <type 'NoneType'>.

Nathan pointed out the reasons of it: [[http://nathanw.net/2013/08/31/pyqgis-dealing-with-null-values/]]

So.. no idea why Postgis and SQLServer are concerned, but we could find a woraround patching mem layer saver to convert NULL to NULL objects.
Transfering to Chris Crook.

Cheers,
Régis

#9 - 2014-02-01 10:54 AM - Chris Crook

Very odd! The MemoryLayerSaver (at least the most recent versions) uses a QDataStream to write the data. This correctly distinguishes between None and NULL. For example see the following code

```
from PyQt4.QtCore import *

file=QFile('/tmp/nulltest')
file.open(QIODevice.WriteOnly)
ds=QDataStream(file)
ds.setVersion(QDataStream.Qt_4_5)
ds.writeQVariant(None)
ds.writeQVariant(NULL)
ds.setDevice(None)
ds=None
file.close()

file.open(QIODevice.ReadOnly)
ds=QDataStream(file)
ds.setVersion(QDataStream.Qt_4_5)

v1=ds.readQVariant()
v2=ds.readQVariant()
print type(v1),v1
print type(v2),v2
```

This returns

```
<type 'NoneType'> None
<class 'PyQt4.QtCore.QPyNullVariant'> NULL
```

So it correctly identifies None and NULL as different.

On the other hand the python interface doesn't distinguish - None and NULL both get stored in the feature as NULL.   As in this code

```
vl=QgsVectorLayer('POINT?field=f1:integer&field=f2:real&field=f3:string(20)','Test','memory')
dp=vl.dataProvider()
fields=dp.fields()
feat=QgsFeature(fields)
for i in ('f1','f2','f3'):
   feat[i]=NULL
   v1=feat[i]
   print type(v1),v1
   feat[i]=None
   v2=feat[i]
   print type(v2),v2
```

which returns

```
<class 'PyQt4.QtCore.QPyNullVariant'> NULL
<class 'PyQt4.QtCore.QPyNullVariant'> NULL
<class 'PyQt4.QtCore.QPyNullVariant'> NULL
<class 'PyQt4.QtCore.QPyNullVariant'> NULL
<class 'PyQt4.QtCore.QPyNullVariant'> NULL
<class 'PyQt4.QtCore.QPyNullVariant'> NULL
```

So the way it looks is that whether there is a None or NULL in the QgsFeature originally, when it is restored through the memory layer saver and installed into the QgsFeature it turns to NULL.

Which doesn't make sense with what you are saying, that NULLs are restored as None. On the other hand it does mean that Nones are restored as

NULL.

I will fix this in the Memory layer saver - hopefully that will also sort out the labelling problem, even thought it looks like it is the wrong way around. It does look like there may be an issue with the QgsFeature SIP wrapping

#10 - 2014-06-04 03:16 AM - Giovanni Manghi

- Subject changed from Data Defined Label Locations NULL value doesn't work as expected with SQL Server Layer to Data Defined Label Locations NULL value doesn't work as expected with SQL Server Layer/PostGIS

- Priority changed from Normal to Severe/Regression

- Target version changed from Future Release - High Priority to Version 2.4

Tagging this as a regression because is ok on 2.0.1.

It seems that with postgis any decimal datatype does not work for label data defined position.

#11 - 2014-06-04 03:16 AM - Giovanni Manghi

- Status changed from Feedback to Open

#12 - 2014-06-10 11:07 PM - Martin Dobias

- Assignee set to Martin Dobias

#13 - 2014-06-10 11:42 PM - Martin Dobias

- Status changed from Open to Closed

Fixed in changeset commit:"863a83b3077ce6cf5c12086a95e4f4b10fd83dcb".